PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 7 :
H04L 9/32, 9/30

A1

(11) International Publication Number: WO 00/49768

(43) International Publication Date: 24 August 2000 (24.08.00)

(21) International Application Number: PCT/IB99/00281

(22) International Filing Date: 17 February 1999 (17.02.99)

(71)(72) Applicant and Inventor: MITTELHOLZER, Thomas [CH/CH]; Flühgasse 33 b, CH-8008 Zürich (CH).

(74) Agent: E. BLUM & CO.; Vorderberg 11, CH-8044 Zürich (CH).

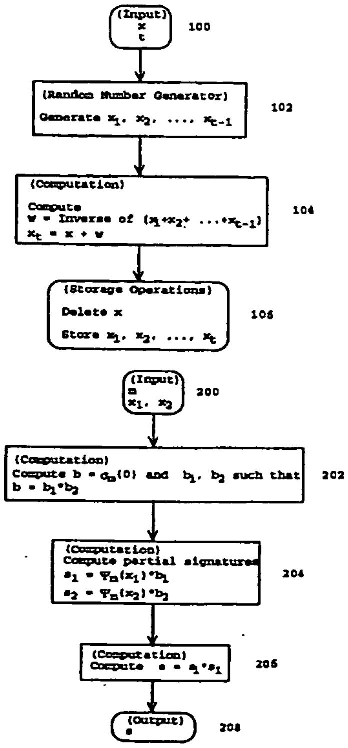
(81) Designated States: European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: METHOD FOR SIGNATURE SPLITTING TO PROTECT PRIVATE KEYS

(57) Abstract

A method for splitting digital signature algorithms is described that can increase the protection of the private key x of the user of an asymmetric key pair (x, y). In an initialization phase, the private key is split into private subkeys. The actual signature splitting method consists of two steps. In a first step (204), partial signature values are computed from the message m to be signed and the subkeys without using the initial private key x. In a second step (206), these partial signature values are combined to form the complete digital signature. To increase the security of the private key x, the private subkeys and the algorithms to compute the partial signature values can be stored and implemented on separate tamper-resistant devices. When a proper subset of the private subkeys becomes compromised, new private subkeys can be generated without having to change the original key pair (x, y).


(Input)
x
t
100
(Random Number Generator)
Generate $x_1, x_2, \ldots, x_{t-1}$
102
(Computation)
Compute
w = Inverse of $(x_1+x_2+\ldots+x_{t-1})$
$x_t = x + w$
104
(Storage Operations)
Delete x
Store $x_1, x_2, \ldots, x_t$
106
(Input)
m
x_1, x_2
200
(Computation)
Compute $b = \sigma_m(0)$ and b_1, b_2 such that $b = b_1 * b_2$
202
(Computation)
Compute partial signatures
$s_1 = \Psi_m(x_1) * b_1$
$s_2 = \Psi_m(x_2) * b_2$
204
(Computation)
Compute $s = s_2 * s_1$
206
(Output)
s
208

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania
AM	Armenia
AT	Austria
AU	Australia
AZ	Azerbaijan
BA	Bosnia and Herzegovina
BB	Barbados
BE	Belgium
BF	Burkina Faso
BG	Bulgaria
BJ	Benin
BR	Brazil
BY	Belarus
CA	Canada
CF	Central African Republic
CG	Congo
CH	Switzerland
CI	Côte d'Ivoire
CM	Cameroon
CN	China
CU	Cuba
CZ	Czech Republic
DE	Germany
DK	Denmark
EE	Estonia
ES	Spain
FI	Finland
FR	France
GA	Gabon
GB	United Kingdom
GE	Georgia
GH	Ghana
GN	Guinea
GR	Greece
HU	Hungary
IE	Ireland
IL	Israel
IS	Iceland
IT	Italy
JP	Japan
KE	Kenya
KG	Kyrgyzstan
KP	Democratic People's Republic of Korea
KR	Republic of Korea
KZ	Kazakstan
LC	Saint Lucia
LI	Liechtenstein
LK	Sri Lanka
LR	Liberia
LS	Lesotho
LT	Lithuania
LU	Luxembourg
LV	Latvia
MC	Monaco
MD	Republic of Moldova
MG	Madagascar
MK	The former Yugoslav Republic of Macedonia
ML	Mali
MN	Mongolia
MR	Mauritania
MW	Malawi
MX	Mexico
NE	Niger
NL	Netherlands
NO	Norway
NZ	New Zealand
PL	Poland
PT	Portugal
RO	Romania
RU	Russian Federation
SD	Sudan
SE	Sweden
SG	Singapore
SI	Slovenia
SK	Slovakia
SN	Senegal
SZ	Swaziland
TD	Chad
TG	Togo
TJ	Tajikistan
TM	Turkmenistan
TR	Turkey
TT	Trinidad and Tobago
UA	Ukraine
UG	Uganda
US	United States of America
UZ	Uzbekistan
VN	Viet Nam
YU	Yugoslavia
ZW	Zimbabwe

Method for Signature Splitting to Protect Private Keys

Technical Field

The present invention relates to a method and apparatus for generating a digital signature according to the preamble of the independent claims.

Background Art

In a Public-Key Crypto System (PKCS) each user has one or more key pairs (x,y) consisting of a private key x and a corresponding public key y (cf. Handbook of Applied Cryptography by A.J. Menezes, P.C. van Oorschot and S.A. Vanstone, CRC Press, 1997, ISBN 0-8493-8523-7). The public key is made available to all users of the PKCS in such a way that the authenticity of the link between a user - which is characterized by a distinguished name - and his public key is guaranteed. The private key x, however, is kept secret and only the authorized user has access to x.

Signature schemes that rely on a PKCS are e.g. RSA (cf. US 4 405 829), or ElGamal based signature schemes, such as the schemes of Schnorr (US 4 995 082) and Nyberg-Rueppel (cf. K.Nyberg, R.Rueppel, "Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem," Designs, Codes and Cryptography, 7, 1996, pp. 61 - 81) or the DSA, see FIPS 186 ("Digital Signature Standard", Federal Information Processing Standards Publication 186, U.S. Department of Commerce/N.I.S.T., National Technical Information Service, Springfield, Virginia, 1994). These digital signature schemes provide methods for signing a digital message and verifying a digital signature. But they do not provide means for protecting the private key.

If an unauthorized party obtains a copy of the private key x, this party can form digital signatures and act as if it were the authorized user. Thus, it is crucial to securely protect the private key x and to avoid that x becomes compromised, e.g., by falling into the hands of an unauthorized party.

The private key is usually protected by an access control system. In a simple access control system, the private key x is stored in encrypted format on a storage device and the private key is only made available if the correct password is provided. The security of an access control system depends on different factors such as the particular access control mechanism, the encryption algorithm used, the device that performs encryption and decryption, and the storage device on which the private key is stored. Possible storage devices could be a diskette, a dedicated protected computer system or a tamper-resistant device such as a chip card or an electronic wallet but also a PC at home.

There are different ways how the private key of a user can become compromised. The following threats may arise.

(I) The access control is compromised. E.g., an unauthorized party has obtained the password or succeeds to read the private key from the storage device.

(II) An authorized party is able to extract (parts of) the private key during the digital signature process from the device that performs the signature.

(III) Information about the private key leaks out to an unauthorized party during the initialization and key distribution phase.

(IV) The underlying PKCS and the corresponding digital signature scheme are broken.

Disclosure of the Invention

The problem to be solved by the present invention is to increase the protection of the private key against at least one of the threats (I)-(III). This problem is solved by the method and apparatus according to the independent claims.

The invention can in particular be used to increase protection against threats (I) and (II). It can also partly increase the protection against threat (III) depending on the key generation and key distribution model.

The present invention makes use of a particular idea from Secret Sharing (cf. Chapter 12.7 in the textbook cited above), viz., the private key is split into two or more private subkeys. In contrast to Secret Sharing, the private subkeys need not be distributed to different entities; in the present invention, the private subkeys can also be managed and used by the same entity. Thus, this invention is based on a different trust model than the one in Secret Sharing. Another important difference to Secret Sharing consists in the way that the subkeys are used. In the present invention, the subkeys need not be communicated to a dedicated entity to form the original private key x; instead, the subkeys are used to create partial signatures and these partial signatures are combined to form the full signature. Thus, when producing a digital signature, the private key x is never generated from the private subkeys. Moreover, the private subkeys cannot be effectively determined from the partial signatures and, hence, even if an unauthorized party knows all partial signatures, the private key is not compromised.

Brief Description of the Drawings

The invention will be better understood and objects other than those set forth above will become apparent when consideration is given to the following detailed description thereof. Such description makes reference to the annexed drawings, wherein:

Fig. 1 shows the steps of the subkey generation phase for generating t private subkeys

Fig. 2 shows the steps of the Signature Splitting Method using t=2 private subkeys

Fig. 3 shows a possible hardware implementation for a signature splitting scheme with t=2 private subkeys.

Modes for Carrying Out the Invention

The present invention provides a method to split digital signatures into partial signatures and to combine these to generate the full original signature. The resulting scheme will be called a Signature Splitting Scheme (SSS).

As a prerequisite, it is assumed that the private key x can be viewed as an element of a group X with group operation +, where 0 denotes the neutral element, and that the signature or a characteristic value s of the signature lies in a monoid S with composition law *. Fixing a message m to be signed, the signature algorithm Σ defines a mapping σ_m from the key group X to the signature monoid S, namely, $s=\sigma_m(x)$, where s is the signature value that results from applying the signature algorithm Σ to m using the private key x. It is further assumed that, for almost all allowed messages m, the mapping ψ_m defined by

$$\psi_m(x) = \sigma_m(x) * (\sigma_m(0))^{-1}, \qquad (1)$$

where $(\sigma_m(0))^{-1}$ denotes the inverse of $\sigma_m(0)$, is a homomorphism from X to S.

In an initialization phase, which will be called Subkey Generation Phase, the private key x is split into two or more private subkeys $x_1, x_2, \ldots$ using a Shared Control Scheme as described in Chapter 12.7.1 in the textbook cited above. A splitting into t private subkeys is obtained by choosing t-1 uniformly random subkeys $x_1, x_2, \ldots, x_{t-1}$ in the group X and by requiring that the last private subkey x_t satisfies the equation

$$x = x_1 + x_2 + \ldots + x_t \ . \qquad (2)$$

The private subkeys are separately stored and protected by separate access control systems. This concludes the initialization phase of the subkey generation.

The signature splitting method makes use of the homomorphism property

$$\psi_m(x) = \psi_m(x_1) * \psi_m(x_2) * \ldots * \psi_m(x_t) \ . \qquad (3)$$

The following steps are carried out:

(i) For a message m to be signed, the value $b = \sigma_m(0)$, which is independent of x, is split into t subvalues $b_1, b_2, \ldots, b_t$ using a pre-defined splitting rule such that in the monoid S the following equation holds

$$b = b_1 * b_2 * \ldots * b_t \ . \qquad (4)$$

(ii) Using the private subkeys, the message m and the previously computed subvalues b_i, the partial signature values

$$s_i = \Psi_m(x_i) * b_i \tag{5}$$

are computed for i=1,2,...,t.

(iii) Eventually, the partial signatures values are combined to form the signature value s, given by

$$s = s_1 * s_2 * \ldots * s_t . \tag{6}$$

Detailed Description for the Implementation of Signature Splitting Schemes

The goal of a SSS is to increase the protection of the private key x. To increase the protection against threats (I) and (II), the private subkeys x_i and the algorithms for the computation of the partial signature values s_i can be stored and implemented on separate tamper-resistant devices, which are under the control of the authorized user of the key pair (x,y). The combining operation (6), in the last step, can be performed on a dedicated device that reads in the partial signature values and generates the output s. This dedicated device need not be necessarily under the control of the authorized user; the combining operation can e.g. take place on the device of the receiver of the digital signature.

A possible hardware implementation of a SSS is shown in Fig. 3 where t=2 private subkeys are used. In the key generation phase, the key pair (x, y) can be generated on a computer (shown as device 300 in Fig. 3). This computer can also contain a program that executes the steps of the Subkey Generation Phase as described above and illustrated in Fig. 1. E.g., the storing operation at step 106 in Fig. 1 will put the private subkeys x_1 and x_2 on the two separate chip cards 304 and 308 shown in Fig. 3.

Suppose a message m obtained via the input interface 310 (e.g. a keyboard) or via the network is to be signed by the user with key pair (x,y) using the computer 300 and the two chip cards 304 and 308, which carry the two private subkeys x_1 and x_2. The digital signature is performed by applying the steps of the signature splitting method described above and illustrated in Fig. 2. The mentioned computer sends the message m to the processors on the two chip cards 304, 308. In order to activate the partial signature computation (step 204 in Fig. 2) on the chip cards 304, 308, the user must enter the two passwords for the two subkeys, which can be done via the keyboard of the computer 300 or via two separate mini-keyboards that are installed on the chip cards or on the two chip card readers. After performing the computation of the subvalues (step 202) and the computation of the partial signatures (step 204), the two chip cards transfer the resulting partial signatures values s_1 and s_2 to the mentioned computer. On this computer, the partial signatures values are combined to the signature value s and completed to the full signature in an appropriate format. It can then be transferred over a network 312 to a computer 314 of another user of the PKCS.

Key Protection and Subkey Re-Generation

Once the subkey generation is completed and all subkeys are stored on dedicated devices, the initial private key x need not be kept and stored in a SSS. Without private key x, direct attacks against the private key are no longer possible. Thus, in a SSS the private key can only be attacked via attacks against the subkeys. The Shared

Control Scheme described above has the following security feature: If the private key x is split into t private subkeys as specified in the initial Subkey Generation Phase, then x will not be compromised unless all t private subkeys are compromised because fewer than t subkeys give no information about the private key x. Thus, if the t subkeys are all stored on separate devices, it is about t times more difficult to obtain all subkeys than it would be to obtain the original private key, when no SSS is used. Therefore, a SSS can increase the protection against threat (I) by about a factor of t. A similar increase of the security of the private key x against threat (II) by a factor of t is obtained if all partial signatures values s_1, s_2, ..., s_t are computed on t separate devices.

If in a digital signature scheme the private key gets compromised, there is no way to recover without replacing the old key pair (x,y) by a new key pair (x',y'). This may have far reaching implications if the user of this key pair represents a particular trustworthy authority such as a certification authority of a public key infrastructure. When an SSS is used, such a mandatory replacement of the private key x can be circumvented provided that not all subkeys have been compromised. The following method for recovering from a partially compromised SSS by re-generation of new subkeys can be applied.

Suppose that the private subkeys x_{i_1}, x_{i_2}, ..., x_{i_u}, where u<t are compromised and that there exists a non-compromised private subkey x_k. The SSS is fully recovered by re-generating u+1 new subkeys x'_{i_1}, x'_{i_2}, ..., x'_{i_u}, x'_k, where u of these new subkeys are chosen uniformly random in the group X and the last new subkey is determined by the equation

$x'_{i_1} + x'_{i_2} + \ldots + x'_{i_u} + x'_k = x_{i_1} + x_{i_2} + \ldots + x_{i_u} + x_k$. (5)

This re-generation method can also be used to exchange a subset of the private subkeys if such a subkey replacement is required by a key management policy.

Signature Splitting for the RSA Signature

The Rivest-Shamir-Adleman (RSA) PKCS is based on the difficulty of factoring a product $n=p\cdot q$ of two large prime numbers p and q (cf. US 4 405 829). Let $Z_{\varphi(n)}$ denote the ring of integers modulo $\varphi(n)$, where $\varphi(n)=(p-1)(q-1)$. The private key x is a randomly chosen invertible element of $Z_{\varphi(n)}$ and the public key is given by n and the inverse y of x, i.e., y satisfies $x\cdot y=1 \mod \varphi(n)$. The key group X consists of the additive group of $Z_{\varphi(n)}$, the signature monoid S consists of the multiplicative structure of the ring Z_n and for a given message m in Z_n, the mapping σ_m is defined by

$$\sigma_m(x) = m^x \mod n \; .$$

In particular, $\sigma_m(0)=1$ and, therefore, the mapping ψ_m defined in (1) coincides with σ_m. This allows to simplify the signature splitting method by skipping the splitting step of the value $b=\sigma_m(0)$ as given in (4). Note that $\psi_m=\sigma_m$ is a homomorphism if and only if m is relatively prime to n, which is true for almost all m. If m is not relatively prime to n, then m can be used to break this RSA PKCS, i.e., an attacker can factor n efficiently. But even in the case that m is not relatively prime to n, the splitting scheme still functions properly, i.e., (3) always holds for every splitting of x as given in (2) because x is relatively prime to $\varphi(n)$.

Signature Splitting for the ElGamal Signature and the DSA

ElGamal based signature schemes rely on the difficulty of the discrete logarithm problem (cf. Chapter 11.5 in the textbook cited above). In the original ElGamal signature scheme, a large finite field GF(q) and a primitive element ß of GF(q) are given. Each user randomly chooses his private key x in the additive group of $X = Z_{q-1}$ and forms his public key $y=ß^x$ in GF(q). Let h denote a suitable hash function and let h(m), $0 \leq h(m) < q-1$, denote the hash value of a message to be signed. The signature for m, consisting of the pair (r,s), is obtained by carrying out the following steps.

(a) Compute $r=ß^k$ in GF(q), where k is a randomly chosen element of Z_{q-1}, which is relatively prime to q-1.

(b) Solve for s in the congruence

$$h(m) = x \cdot h(r) + k \cdot s \bmod (q-1).$$

The signature value s lies in the signature monoid $S = Z_{q-1}$, which is actually a group. The signature mapping σ_m is given by

$$s = \sigma_m(x) = k^{-1} \cdot (h(m) - x \cdot h(r))$$

and the message dependent value b equals $\sigma_m(0)=k^{-1} \cdot h(m)$. In an ElGamal based SSS, step (a), which does not depend on the private key x, is performed as in the ElGamal scheme and the signature splitting is applied to step (b). In this setting, where X = S, a possible splitting rule for the message dependent value b is given by the splitting rule for the private subkeys as specified in the Subkey Generation Phase.

The DAS of the DSS as described in FIPS 186 ("Digital Signature Standard", Federal Information Processing Standards Publication 186, U.S. Department of Commerce/N.I.S.T., National Technical Information Service, Springfield, Virginia, 1994) is based on the ElGamal scheme. For the DSA it is assumed that q is a large prime and that there is a prime u in the range $2^{159} < u < 2^{160}$, which is a divisor of q-1. Moreover, $\beta \in GF(q)$ is assumed to be a generator of the unique cyclic subgroup of order u in the multiplicative group of GF(q). Similarly as in the ElGamal scheme, the signature of a message m consists of the pair (r, s), where

$$r = (\beta^k \bmod q) \bmod u$$

and

$$s = k^{-1} (h(m) + x \cdot r) \bmod u.$$

Hence, the signature splitting can be carried out in a similar way as in the ElGamal scheme.

Signature Splitting for the Schnorr Signature

The Schnorr signature scheme (US 4 995 082) is a variant of the ElGamal scheme. As a new idea, instead of being a primitive element in GF(q), ß is now a generator of a large subgroup of the multiplicative group of GF(q). Thus, ß generates a group isomorphic to Z_u, where u divides q-1. The key pair (x,y) is defined as above, i.e, $y=ß^x$ where x is an element of the key group $X = Z_u$. Moreover, to reduce the message length a hash function h is used.

The signature for m, consisting of the pair (e,s), is obtained by carrying out the following steps.

(a') Compute $r=ß^k$ in GF(q), where k is a randomly chosen element of Z_u.

(b') Form the concatenation $m||r$ of m and r and compute the hash value $e=h(m||r)$.

(c') Compute the signature value

$$s = \sigma_m(x) = x \cdot e + k \bmod u \ .$$

The signature value s lies in the signature monoid $S = Z_u$, which is actually a group. The value b equals $\sigma_m(0)=k$ and, thus, does not depend on m. This value can be split into subvalues $b_i=k_i$ using the method of the Subkey Generation Phase for the group $S = Z_u$. Since k is random, one can generate this random value by randomly selecting the subvalues k_i and by setting

$$k = k_1 + k_2 + \ldots + k_t \ . \qquad (7)$$

In a Schnorr based SSS, the splitting method can be applied to step (a'), i.e., one computes the pairs (k_i,r_i) separately, where $r_i=ß^{ki}$ for i=1,2, ..., t. To carry out step (b), one needs only the values r_i and the message m as input. The hash value e is computed as above using the product $r = r_1 \cdot r_2 \cdot \ldots \cdot r_t$ (in GF(q)). In step (c'), the partial signature values $s_i=x_i \cdot e+k_i$ mod u are computed separately before they are combined to form the signature value s.

Note that in this Schnorr based SSS, the random elements k_i can be generated and kept on the same separate storage and computing devices as the private subkeys x_i and these elements never need to leave these separate devices.

<u>Signature Splitting for the Nyberg-Rueppel Signature</u>

The Nyberg-Rueppel signature scheme (cf. K.Nyberg, R.Rueppel, "Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem," Designs, Codes and Cryptography, 7, 1996, pp. 61 - 81) is another variant of the ElGamal scheme, where GF(q) is a prime field, i.e., q is a prime. As in the Schnorr scheme, the key group X consists of a large subgroup Z_u, where u divides q-1. The key pair (x,y) is defined as in the Schnorr scheme. Instead of a hash function, a redundancy function ρ is used, which is applied to a set of allowed messages. A message m from this set is signed by carrying out the following steps.

(a'') Compute the redundancy value $m'=\rho(m)$.

(b'') Compute $r=ß^{-k}$ in GF(q), where k is a randomly chosen element of Z_u.

(c'') Compute $e=m'\cdot r$ in GF(q).

(d'') Compute the signature value

$$s = \sigma_m(x)= x\cdot e + k \mod u \ .$$

The signature consists of the pair (e,s). In a Nyberg-Rueppel based SSS, step (a'') is performed as is. The splitting method is applied to both step (b'') and (d''). In step (b''), one uses the splitting method for the random element k as described in the Schnorr based SSS (cf. equation (7)) and generates the pairs (k_i, r_i), where $r_i=ß^{-ki}$. In Step (c''), the value e is computed from m and the values r_i using the product $r = r_1\cdot r_2\cdot\ldots\cdot r_t$ (in GF(q)). In step (d''), the partial signature values $s_i=x_i\cdot e+k_i \mod u$ are computed separately before they are combined to form the signature value s.

Signature Splitting for Elliptic Curve Based Signatures

ElGamal based digital signatures schemes can also be defined over elliptic curves. Instead of considering the multiplicative group of GF(q), one considers a large cyclic subgroup U of an elliptic curve C, which itself forms a group with additive group operation •. The subgroup U is generated by some generator ß, which is a point of the elliptic curve C. Let u denote the order of the subgroup U. The mapping of the integers Z onto U given by assigning to an integer i the i-fold 'sum' $i \cdot ß=(ß•ß•...•ß)$ induces an isomorphism from Z_u onto U. For ElGamal based digital signature schemes over elliptic curves, one can apply the signature splitting method in a similar way as for the ElGamal based schemes above. E.g., the key group is $X = Z_u$ and the signature monoid S also equals Z_u.

Secret Sharing and Signature Splitting

Instead of using a simple Shared Control Scheme as described above, more general Secret Sharing Schemes can be applied, where a secret x is shared by e.g. 4 persons and whenever 2 of these 4 persons put together their shares x_i, they can reconstruct the secret x. These more general type of Secret Sharing Schemes can be combined with signature splitting if the group operations that are used are compatible with those of the underlying signature scheme.

Consider e.g. an RSA PKCS with $n=p \cdot q$ and a key pair (x,y). The secret sharing scheme given in the first example in the paper "On Secret Sharing" by E.D. Karnin, J.W. Green and M.E. Hellman (in IEEE Trans. on Information Th., Vol. 29, No. 1, Jan 1983, pp. 35 - 41) can be adapted to work for signature splitting. To this end, the condition C3) of the mentioned paper is dropped.

The Subkey Generation Phase consists of two steps. In a first step, the private key x is split into $x=u_1+u_2$ mod $\varphi(n)$. In a second step, the secret (u_1,u_2) is divided into 4 shares

$$\begin{aligned}\mathbf{x}_1 &= (u_3,u_4),\\ \mathbf{x}_2 &= (u_1+u_2+u_3,u_2+u_4),\\ \mathbf{x}_3 &= (u_2+u_3,u_1+u_4),\\ \mathbf{x}_4 &= (u_1+u_3,u_2+u_3+u_4)\end{aligned}$$

where + denotes addition modulo $\varphi(n)$ and where u_3 and u_4 are randomly chosen. Eventually, the 4 shares are stored on separate devices.

For a message m, the signature splitting is characterized by the pairs of partial signature values

$$\mathbf{s}_i = (m^{x_{i1}},\ m^{x_{i2}}) \quad \text{mod } n$$

where x_{i1} denotes the first and x_{i2} the second component of $\mathbf{x}_i$. From any 2 of the 4 partial signature pairs $\mathbf{s}_i$ - when combining their components suitably - one can compute

$$\mathbf{s} = (m^{u1},\ m^{u2}) \text{ mod } n\ .$$

The final signature value is obtained by multiplying the two components of $\mathbf{s}$, i.e., $s=m^{u1}\cdot m^{u2}$ mod n.

The above can be generalized as follows. A t-out-of-w Secret Sharing Scheme, where the secret x is split into w shares x_i lying in a subkey group X' with group operation +', can be characterized by requiring that there exist reconstruction functions $f_{i_1 i_2 \cdots i_t}$ from the t-fold direct product $X' \times X' \times \ldots \times X'$ into the key group X for any t-element subset $i_1,\ i_2, \ldots,\ i_t$ such that $x = f_{i_1 i_2 \cdots i_t}(x_{i_1}, x_{i_2}, \ldots, x_{i_t})$. Suppose that $f_{i_1 i_2 \cdots i_t}$ is a homomorphism and that the partial signature values s_i are contained in a monoid S' with composition law *'. Define a homomorphism $g_{i_1 i_2 \cdots i_t}$ from the t-fold direct product $S' \times S' \times \ldots \times S'$ into the signature monoid S, which is derived from $f_{i_1 i_2 \cdots i_t}$ by replacing the group operations $+'$ and $+$ by the composition laws $*'$ and $*$, respectively. The Secret Sharing Scheme is compatible with the signature scheme if, for almost all messages m, there exists a homomorphism ψ'_m from the subkey group X' to the monoid S' that is compatible with ψ_m, i.e., for every t-tuple $v_1, v_2, \ldots, v_t$ in $X' \times X' \times \ldots \times X'$ the following equation in S must hold

$$\psi_m(f_{i_1 i_2 \cdots i_t}(v_1, \ldots, v_t)) = g_{i_1 i_2 \cdots i_t}(\psi'_m(v_1), \ldots, \psi'_m(v_t)).$$

For such a compatible Secret Sharing Scheme, one can generate the partial signature values $s_i = \psi'_m(x_i) *' b_i$ in the partial signature monoid S', where the b_i, $i=1,2,\ldots,w$, are elements of the partial signature monoid S' such that

$$g_{i_1 i_2 \cdots i_t}(b_{i_1}, b_{i_2}, \ldots, b_{i_t}) = \sigma_m(0).$$

The combining operation, which generates the signature value s out of any t partial signatures, is given by

$$s = g_{i_1 i_2 \cdots i_t}(s_{i_1}, s_{i_2}, \ldots, s_{i_t}).$$

While there are shown and described presently preferred embodiments of the invention, it is to be distinctly understood that the invention is not limited thereto but may be otherwise variously embodied and practiced within the scope of the following claims.

Claims

1. A method for generating a digital signature comprising a signature value $s = \sigma_m(x)$ using a signature algorithm Σ and a private and public key pair x,y for a message m, wherein x is an element of a group X with group operation +, where 0 denotes the neutral element, and the signature value s is an element of a monoid S with composition law * and wherein the map ψ_m defined by $\psi_m(x) = \sigma_m(x) * (\sigma_m(0))^{-1}$ is a homomorphism from X to S for almost all messages m,

said method comprising the steps of

providing $w \geq 2$ private subkeys $x_1, x_2, \ldots, x_w$ in a subkey group X' with group operation +' such that said private key x can be reconstructed from any subset of at least t, $2 \leq t \leq w$, subkeys $x_{i_1}, x_{i_2}, \ldots, x_{i_t}$ using $x = f_{i_1 i_2 \ldots i_t}(x_{i_1}, x_{i_2}, \ldots, x_{i_t})$,

using said subkeys for generating partial signature values $s_i = \psi'_m(x_i) *' b_i$ in a partial signature monoid S'

and generating said signature value s from any t partial signatures using $s = g_{i_1 i_2 \ldots i_t}(s_{i_1}, s_{i_2}, \ldots, s_{i_t})$,

wherein $f_{i_1 i_2 \ldots i_t}$ is a homomorphism from the t-fold direct product $X' \times X' \times \ldots \times X'$ into the key group X and $g_{i_1 i_2 \ldots i_t}$ is a homomorphism from the t-fold direct product $S' \times S' \times \ldots \times S'$ into the signature monoid S, which is derived from $f_{i_1 i_2 \ldots i_t}$ by replacing the group operations +' and + by the composition laws *' and *, respectively,

where the b_i, i=1,2,...,w, are elements of the partial signature monoid S' such that $g_{i_1 i_2 \ldots i_t}(b_{i_1}, b_{i_2}, \ldots, b_{i_t}) = \sigma_m(0)$,

and where, for almost all messages m, ψ'_m is a homomorphism from the subkey group X' to the partial signature monoid S' compatible with ψ_m, i.e., for every t-tuple $v_1, v_2, \ldots, v_t$ in $X' \times X' \times \ldots \times X'$ the following equation in S must hold $\psi_m(f_{i_1 i_2 \ldots i_t}(v_1, \ldots, v_t)) =$

$g_{i_1 i_2 \ldots i_t}(\psi'_m(v_1), \ldots, \psi'_m(v_t))$.

2. The method of claim wherein said signature scheme is the RSA signing algorithm.

3. The method of claim 1 wherein said signature algorithm Σ is the ElGamal, the DSA of the DSS, the Schnorr or the Nyberg-Rueppel signature algorithm over the originally specified groups or over subgroups of an elliptic curve.

4. The method of one of the preceding claims wherein said step of generating said partial signature values $s_1, s_2, \ldots, s_w$ is carried out in a secure environment and said step of generating said signature value from said partial signature values is carried out in a non-secure environment.

5. The method of one of the preceding claims wherein $X'=X$, $S'=S$, $\psi'_m=\psi_m$ and $t=w$ and where $f_{12\ldots t}(x_1, \ldots, x_t) = x_1+x_2+\ldots+x_t$ and $g_{12\ldots t}(s_1, \ldots, s_t) = s_1 * s_2 * \ldots * s_t$.

6. The method of claim 5 comprising the step of generating a new set of subkeys $\{x'_{i_1}, x'_{i_2}, \ldots, x'_{i_u}, x'_k\}$ from said subset and at least one non-compromised subkey x in case that a proper subset $\{x_{i_1}, x_{i_2}, \ldots, x_{i_u}\}$ of said subkeys is compromised or to be replaced.

7. The method of claim 6 wherein said new set of subkeys is generated such that $x'_{i_1}+x'_{i_2}+\ldots+x'_{i_u}+x'_k = x_{i_1}+x_{i_2}+\ldots+x_{i_u}+x_k$.

8. The method of one of the preceding claims comprising the step of storing at least one of said subkeys x_i separately on a tamper-resistant device.

9. The method of claim 8 wherein said tamper-resistant device is a chip card.

10. The method of one of the claims 8 or 9 wherein said step for generating the partial signature values $s_i = \psi'_m(x_i)*b_i$ is carried out in said tamper-resistant device.

11. An apparatus for generating a digital signature comprising means for carrying out the method of one of the preceding claims.

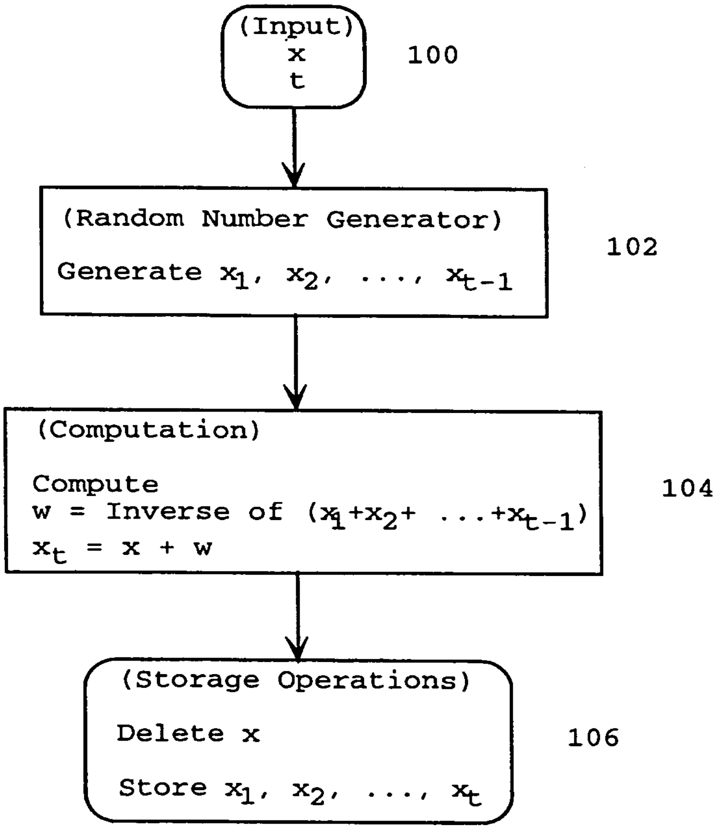

(Input)
x
t
100
(Random Number Generator)
Generate $x_1, x_2, \ldots, x_{t-1}$
102
(Computation)
Compute
w = Inverse of $(x_1+x_2+\ldots+x_{t-1})$
$x_t = x + w$
104
(Storage Operations)
Delete x
Store $x_1, x_2, \ldots, x_t$
106

Fig. 1

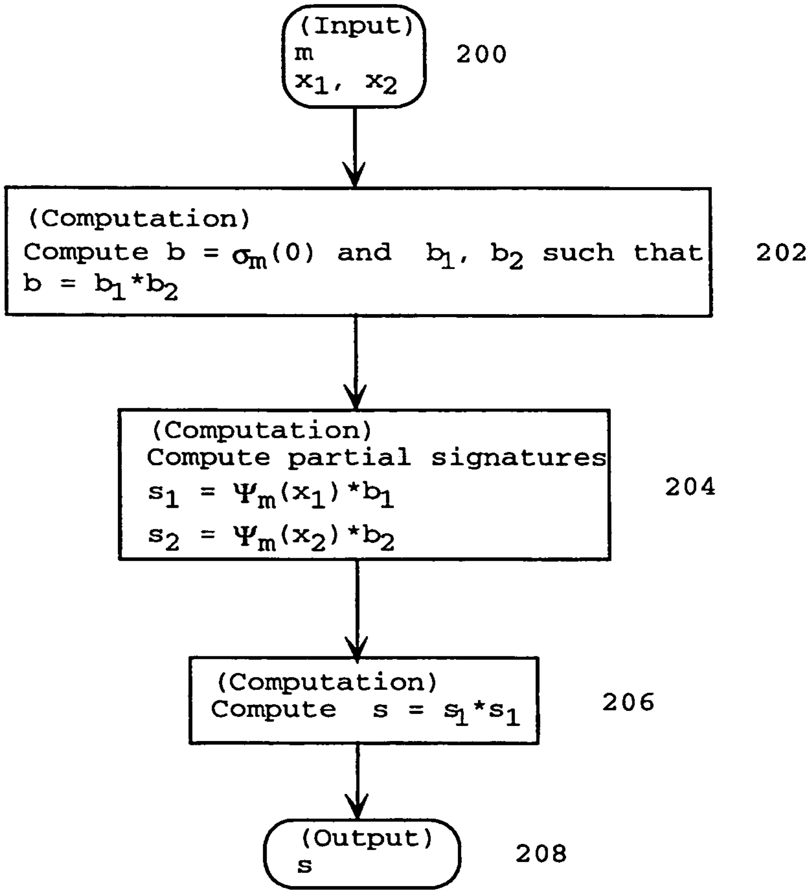

(Input)
m
x_1, x_2
200
(Computation)
Compute $b = \sigma_m(0)$ and b_1, b_2 such that $b = b_1 * b_2$
202
(Computation)
Compute partial signatures
$s_1 = \Psi_m(x_1) * b_1$
$s_2 = \Psi_m(x_2) * b_2$
204
(Computation)
Compute $s = s_1 * s_1$
206
(Output)
s
208

Fig. 2

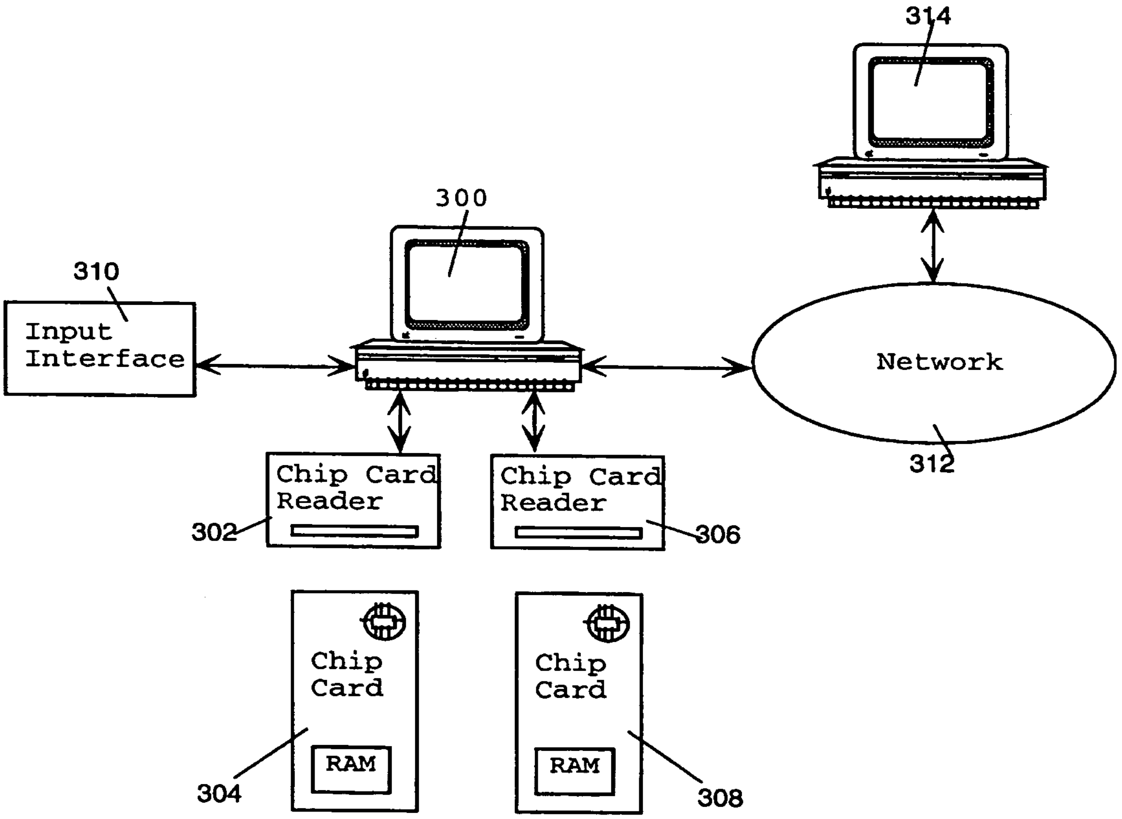

314
300
310
Input Interface
Network
312
Chip Card Reader
Chip Card Reader
302
306
Chip Card
Chip Card
RAM
RAM
304
308

Fig. 3

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/IB 99/00281

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L9/32 H04L9/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5 825 880 A (SUDIA FRANK W ET AL) 20 October 1998 (1998-10-20) abstract column 2, line 21 - line 64	1,4,5,8, 9,11
A	column 5, line 22 -column 6, line 20 claims 1,2 figures 1-5 ---	2,3,6
A	EP 0 869 635 A (FUJITSU LTD ;HITACHI LTD (JP); MAMBO MASAHIRO (JP); OKAMOTO EIJI ()) 7 October 1998 (1998-10-07) abstract page 3, line 1 - line 25 page 5, line 45 -page 6, line 51 claims 1,2 figures 1-3 --- -/--	1-11

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search 19 October 1999	Date of mailing of the international search report 26/10/1999
Name and mailing address of the ISA European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Authorized officer Gautier, L

INTERNATIONAL SEARCH REPORT

Inter: ıal Application No
PCT/IB 99/00281

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	BURMESTER M: "HOMOMORPHISMS OF SECRET SHARING SCHEMES: A TOOL FOR VERIFIABLE SIGNATURE SHARING" ADVANCES IN CRYPTOLOGY - EUROCRYPT '96 INTERNATIONAL CONFERENCE ON THE THEORY AND APPLICATION OF CRYPTOGRAPHIC TECHNIQUES, SARAGOSSA, MAY 12 - 16, 1996, 12 May 1996 (1996-05-12), pages 96-106, XP000725437 MAURER U (ED) ISBN: 3-540-61186-X the whole document -----	1-10

2

INTERNATIONAL SEARCH REPORT

ormation on patent family members

Interr nal Application No

PCT/IB 99/00281

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5825880 A	20-10-1998	AU 5552196 A	24-12-1996
		BR 9608416 A	29-12-1998
		CA 2223305 A	12-12-1996
		CN 1192834 A	09-09-1998
		EP 0872080 A	21-10-1998
		GB 2301919 A	18-12-1996
		JP 11506222 T	02-06-1999
		WO 9639765 A	12-12-1996
		US 5867578 A	02-02-1999
		ZA 9603635 A	19-11-1996
		AP 626 A	16-01-1998
		AU 1680395 A	01-08-1995
		AU 705473 B	20-05-1999
		AU 6084296 A	10-10-1996
		BR 9506414 A	09-09-1997
		CA 2176032 A	20-07-1995
		CN 1138927 A	25-12-1996
		CZ 9601978 A	12-03-1997
		EP 0739560 A	30-10-1996
		HU 75800 A,B	28-05-1997
		JP 9507729 T	05-08-1997
		NZ 279622 A	27-04-1998
		PL 315574 A	12-11-1996
		WO 9519672 A	20-07-1995
		US 5857022 A	05-01-1999
		US 5872849 A	16-02-1999
		US 5850451 A	15-12-1998
		US 5799086 A	25-08-1998
		US 5841865 A	24-11-1998
EP 0869635 A	07-10-1998	JP 10274926 A	13-10-1998